(2)

CDRL No. 0002AB-3

Quarterly Technical Report - Report No. 3
July 1, 1992 - September 30, 1992
DARPA DICE Manufacturing Optimization

Linda J. Lapointe
Thomas Laliberty
Nancy Toro
Robert V.E. Bryant

Raytheon Company

1992

DARPA
Defense Advanced Research
Projects Agency

CDRL No. 0002AB-3

Quarterly Technical Report - Report No. 3
July 1, 1992 - September 30, 1992
DARPA DICE Manufacturing Optimization

Prepared by
Linda J. Lapointe
Thomas Laliberty
Nancy Toro
Robert V.E. Bryant

Raytheon Company
Missile Systems Laboratories
Tewksbury, MA 01876

October 1992

ARPA Order No. 8363/02
Contract MDA972-92-C-0020

Prepared for

DARPA
Defense Advanced Research
Projects Agency

Contracts Management Office
Arlington, VA 22203-1714

Contents

Figures and Tables

1. Summary

This is the Quarterly Technical Report for the DARPA DICE Manufacturing Optimization. The goal of the Manufacturing Optimization (MO) system is to facilitate a two tiered team approach to the product/process development cycle where the product design is analyzed by multiple manufacturing engineers, and the product/process changes are traded concurrently in the product and process domains. The system will support Design for Manufacturing and Assembly (DFMA) with a set of tools to model the manufacturing processes, and manage tradeoffs across multiple processes. The subject of this report is the technical work accomplished during the third quarter of the contract. This report describes the initial MO Prototype, as well as, the evaluations of the Project Coordination Board/Communications Manager (PCB/CM) and Requirements Manager (RM).

Raytheon developed a prototype which demonstrated the first pass at MO functionality. The prototype system was developed to provide a means of assessing the viability and effectiveness of the capabilities defined in the MO functional requirements. The prototype was built by utilizing two existing Raytheon developed systems, RAPIDS–Raytheon Automated Placement and Interconnect System and MOSS–Manufacturing Optimization Support System, the STEP Toolkit (including the ROSE DB system) from STEP Tools Inc., the Requirements Manager from Cimflex Teknowledge, and the Project Coordination Board from CERC. The prototype demonstrates concurrent design, concurrent analysis, design conflict detection, and design change merging of PWB designs. The functionality in the prototype includes the RAPIDS to ROSE translator, the delta file and design merging capabilities of ROSE, a generic difference report generator, and a printed circuit board design flow (task structure) modeled in the Project Coordination Board (PCB). A select set of manufacturing guidelines were modeled in the Requirements Manager (RM) as a standalone application. The prototype was integrated with RAPIDS and MOSS.

Raytheon proposed to use the Project Coordination Board to support the product-to-process team communication throughout the entire product development cycle. As part of the evaluation process and the MO prototype, we modeled and initialized a sample PWB design cycle using the task structure file format, and then simulated a simple two tiered approach to the product/process development activities by stepping through the entire task structure. Based on

the results of our sample test case, we believe that the PCB is able to handle only simple task structures, since the only reliable and complete way to get a task structure into the system is through the use of the flat file format which is cumbersome. At this time, we still feel the system is not mature enough and lacks the functionality required for real-world environments.

Raytheon proposed to use the Requirements Manager to manage and evaluate the manufacturability/producibility guidelines against the product design data. As part of the evaluation process and the MO prototype, we modeled a set of sample manufacturing guidelines along with the corresponding product data structure, and populated the RM database with the model. We then tested the evaluation capabilities of the RM. Based on the results of our evaluation, we believe that the RM is able to provide the capabilities needed by MO for testing product design data for compliance to manufacturing guidelines.

In addition to the MO prototype development efforts, we have been designing an approach for modeling manufacturing processes. The process modeling technique proposed for the MO system is being designed as an AND/OR dependency graph made up of selectable manufacturing processes, which can be either a process, operation, or step object. Each object in the model can be connected to object or objects at a higher and/or lower level in the graph. The difference between the object types is in the level of processing or planning detail defined (i.e. process decisions, operation planning, and/or detail operation planning). Two EXPRESS schemas, which represent the initial efforts on the MO process modeling design, were developed. The first schema is the MO process model. The second schema represents the valid grammar format for the MO process rules required for selecting the processes which are required to manufacture the part.

Raytheon will continue development of MO during the next quarter based on the initial prototype efforts for the Manufacturing Optimization (MO) System developed during the reporting period. Raytheon is also in the process of developing the Design Specification which will be delivered during the next quarter.

2. Introduction

This is the Quarterly Technical Report for the DARPA DICE Manufacturing Optimization. The concept behind the Manufacturing Optimization (MO) system is to facilitate a two tiered team approach to the product/process development cycle where the product design is analyzed by multiple manufacturing engineers, and the product/process changes are traded concurrently in the product and process domains. The system will support DFMA with a set of tools to model the manufacturing processes, and manage tradeoffs across multiple processes. The subject of this report is on the technical work accomplished during the third quarter of the contract. This report describes the development efforts of the initial MO prototype, as well as, the evaluations of the Project Coordination Board/Communications Manager (PCB/CM) and Requirements Manager (RM).

A prototype system was developed to provide a means of assessing the viability and effectiveness of the capabilities defined in the MO functional requirements. The prototype demonstrates an implementation of software tools that support a two-tiered virtual team methodology. The prototype utilizes two existing Raytheon developed systems, RAPIDS–Raytheon Automated Placement and Interconnect System and MOSS–Manufacturing Optimization Support System, the STEP Toolkit (including the ROSE DB system) from STEP Tools Inc., the Requirements Manager from Cimflex Teknowledge, and the Project Coordination Board from CERC. The rest of this report is devoted to methods, assumptions, and procedures, as well as, the results, discussions, and conclusions regarding the prototype and the evaluation of the PCB and RM.

3. MO Prototype

3.1 Methods, Assumptions, and Procedures

3.1.1 Overview

The concept of the Manufacturing Optimization (MO) system is to facilitate a two tiered team approach to the product/process development cycle where the product design is analyzed by multiple manufacturing engineers, and the product/process changes are traded concurrently in the product and process domains. A prototype system has been developed which provides a means of assessing the viability and effectiveness of the capabilities defined in the MO functional requirements. The prototype was built by utilizing two existing Raytheon developed systems, RAPIDS–Raytheon Automated Placement and Interconnect System and MOSS–Manufacturing Optimization Support System, the STEP Toolkit (including the ROSE DB system) from STEP Tools Inc., the Requirements Manager, and the Project Coordination Board. The prototype is intended to demonstrate concurrent design, concurrent analysis, design conflict detection, and design change merging of PWB designs.

RAPIDS is Raytheon's conceptual design and analysis workstation for Printed Wiring Boards (PWB). RAPIDS supports component placement and placement density analysis, as well as a number of other analysis functions, including automatic component insertion checking. Interfaces between RAPIDS and the PWB analysis tools for the following criteria are also provided as part of the RAPIDS tool suite:

- Manufacturing
- Post Layout Effects
- Reliability
- Thermal

At Raytheon, RAPIDS is used for conceptual design and analysis of PWB's. RAPIDS serves in the same capacity at Raytheon that many commercial CAD systems (e.g. Mentor Board Station, Racal-Redac Visula, Cadence, etc.) are used in at other companies. RAPIDS provides an Application Procedural Interface (API) with its database. This enables RAPIDS to be easily interfaced with other systems and standards. Using RAPIDS put the prototype implementation inline with Raytheon methodologies, but does not exclude interfacing MO with

commercially available CAD systems in the future. The key to interfacing MO with a large base of CAD systems is the utilization of the STEP standard by the commercial CAD industry.

MOSS is an internally developed Raytheon system used for PWB fabrication and assembly cost and yield estimations. MOSS analyzes the design features of the PWB and generates a cost and yield estimate based on a historical model of fabricating and assembling similar designs. MOSS provides design guidance based on established standards and practices. Design tradeoffs can be made, the analysis rerun, and the results compared to previous runs.

The prototype developed will be used to model the two tiered team approach where RAPIDS will serve as the tool that the product virtual team uses for conceptual design. Information from the lower level "specialized" process teams will be supplied concurrently. MOSS will be used by the "specialized" process team for manufacturability analysis. Within manufacturing, MOSS will be utilized concurrently by engineers responsible for PWB fabrication planning and Circuit Card Assembly planning. Recommendations will be compared and negotiated among the individual manufacturing participants. After the manufacturing team has reached a consolidated position, the results are passed back to the cross functional (top level) team for their negotiation. The Project Coordination Board will support the communication of the product-to-process team throughout the entire product development cycle. Figure 5-1 shows which functional areas of the two level team will use RAPIDS and/or MOSS.

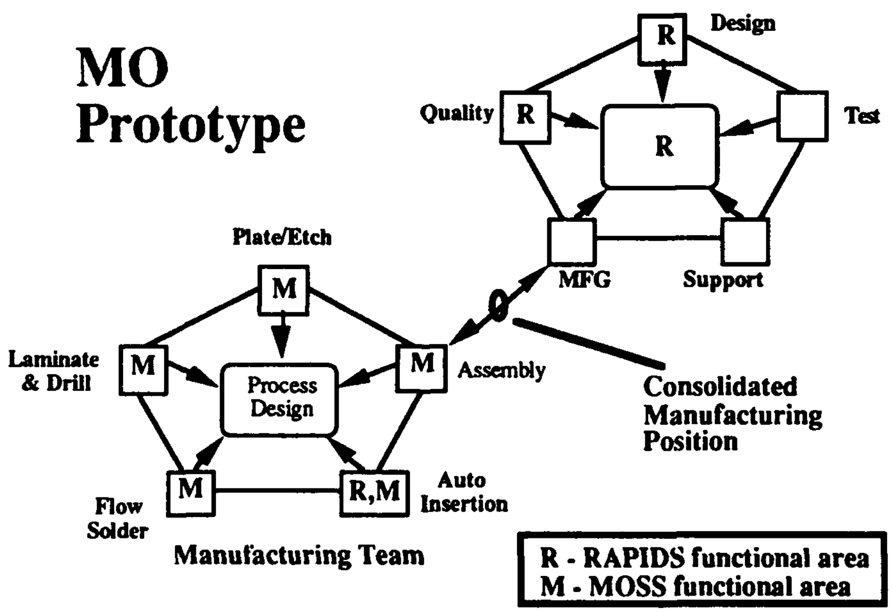

Figure 3-1. Two-Tiered Virtual Team Concept for the MO Prototype

The MO prototype can be used for concurrent design, concurrent analysis, design conflict detection, and design change merging of PWB designs. The prototype environment provides for a PWB design to be initiated in RAPIDS by the top level team. Product engineers determine the board outline profile. An Electrical Engineer (EE) uses RAPIDS for a component fitting study. If the components fit, the EE performs the initial component placement. When placement is complete, the product engineer uses the density analysis tools provided by RAPIDS and the electrical requirements provided by the EE to determine the initial layer stackup.

Once the board profile, layer stackup, and initial placement are defined, the design is passed to the second level teams for their analysis and input. Design, Test, Quality, Support, and Manufacturing process teams would concurrently analyze and provide input in the form of consolidated team positions. The consolidated positions are passed back to the top-level team for conflict detection and design merging.

The MO prototype will address the two-tier manufacturing virtual team. RAPIDS will be used by the Manufacturing team to perform automatic component insertion checking. RAPIDS uses actual data, which contains models of insertion machines from the manufacturing enterprise and models of the physical characteristics of the components in the design, to determine if the components can be automatically inserted. Changes in component packages, component positions, and/or rotations may be recommended by the Manufacturing team and passed back to the top-level team.

The individual process engineers (plate/etch, laminate and drill, flow solder, and assembly) will use MOSS to perform cost and yield analyses on the design. MOSS also uses a model of the Manufacturing enterprise including the cost and yield models of the various manufacturing processes. These model are based on historical data and are continually updated. Change recommendations will be made to the design by each engineer on the Manufacturing Team in order to optimize the board for the process which he/she is responsible for. Conflicts must be identified next. This is done using software provided by STEP Tools Inc. Once the conflicts have been resolved, a consolidated manufacturing position can be passed to the top level team.

Both the Top Level Product Team and Manufacturing Team will be using the Requirements Manager throughout the product development cycle to evaluate the extent to which the product design data meets the specified manufacturability/producibility guidelines.

3.1.2 MO Prototype Architecture

A diagram of the MO prototype architecture is shown in Figure 3-2. The MO prototype components are summarized by function in Table 3-1.

Table 3-1. Summary of MO Prototype Components

MO Prototype Component	Function
RAPIDS	Raytheon Automated Placement and Interconnect System. PWB conceptual design and analysis workstation.
MOSS	Manufacturing Optimization Support System. PWB design cost and yield analysis tool.
rapids_to_step	Interface which translates a RAPIDS database into a STEP Part 21 physical file. The physical file is used for information exchange between top level product team and lower level process teams.
step_to_rapids	Interface which translates a STEP Part 21 physical file into a RAPIDS database. This necessitated since the top-level team uses RAPIDS as the conceptual design system.
STEP Tools diff	The *diff* tool provided by STEP Tools Inc. will compare two versions of a design stored in STEP. It will produce a delta file which when applied to design version one will make it equivalent to version two.
STEP Tools sed	The *sed* tool provided by STEP Tools Inc. will apply a delta file created by the *diff* tool when comparing two STEP designs. When applied, the design it was run on will be equivalent to the one that it was compared to.
Difference Report Generator	The Difference Report Generator reads a delta file created by the *diff* tool and a design stored in STEP. The report generator presents the changes that will be made if the delta file is applied to the design in a concise format.
Project Coordination Board	The PCB provides support for the coordination of the product development activities in a cooperative environment. It provides common visibility and change notifications. It is being used in the MO Prototype to support communication of product/process development activities.
Communications Manager	The CM supports communication among team members, and its supports distributed computing and database access in a network. The CM is part of the MO prototype because it is required to use the PCB.
Requirements Manager	The RM is a tool designed to manage product requirements and evaluate the compliance of product design data with requirements. It is being used in the MO Prototype to manage the manufacturability/producibility guidelines.

The Project Coordination Board provides support for the coordination of the product development activities in a cooperative environment. It provides common visibility and change

notifications. It is being used in the MO Prototype to support the communication of product/process development activities (refer to Section 4).

The top level team uses RAPIDS. At Raytheon, PWB data is stored in the RAPIDS database. The PWB design is passed to the second tier teams (in this case the Manufacturing Team) using a STEP physical file.

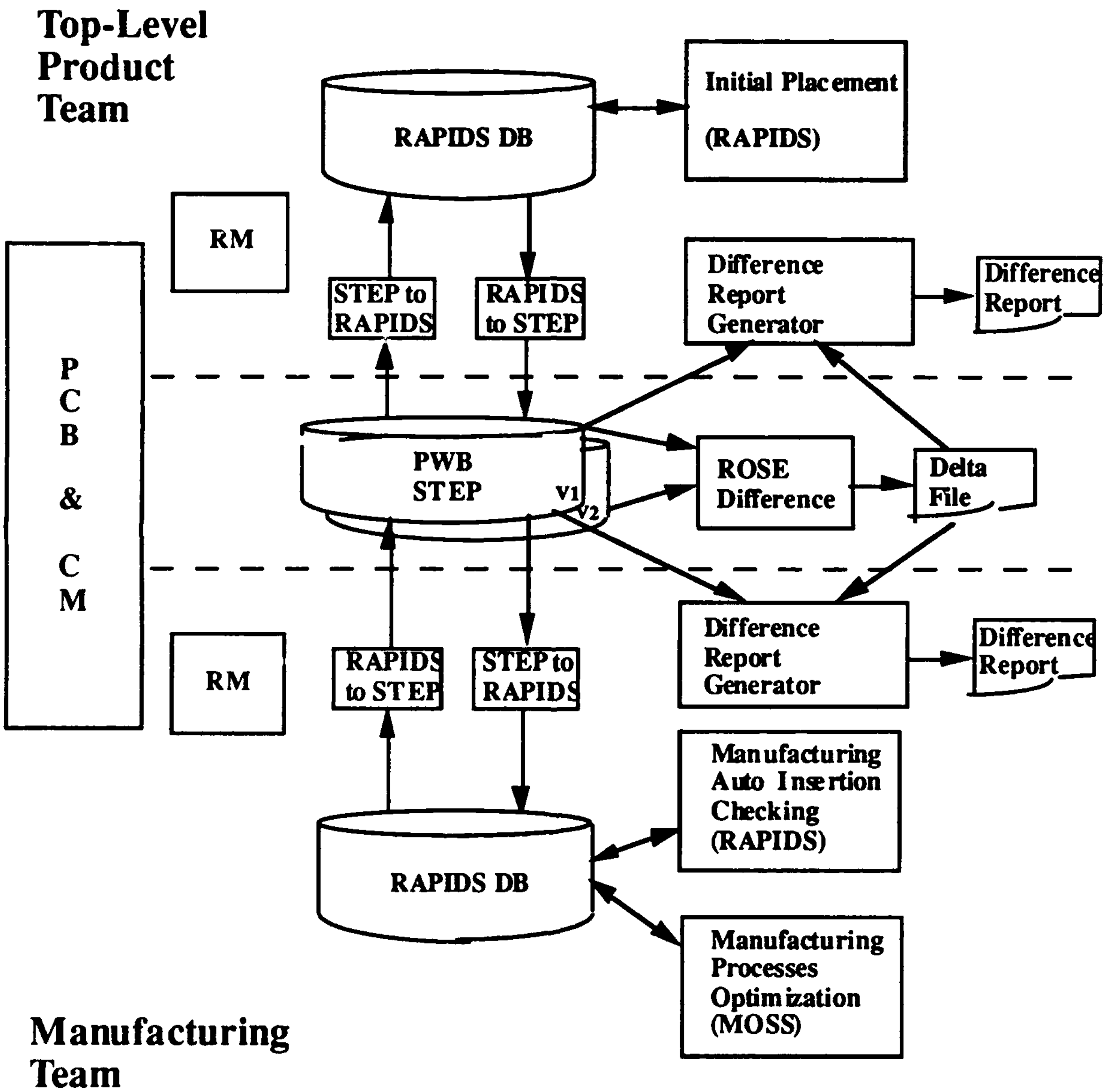

Figure 3-2. MO Prototype Architecture

Generating the STEP physical file is facilitated by the interface *RAPIDS to STEP* which maps RAPIDS data items into instantiated STEP entities. We created an information model

using the EXPRESS information modeling language. The model was based on the RAPIDS database. The EXPRESS information model was compiled using the STEP Tools *express2c++* compiler which generated a STEP schema and a C++ class library. The class library consists of methods for creating and referencing persistent instances of the STEP entities which are stored in a ROSE database. The STEP schema is used by the STEP Tools *STEP filer* for reading and writing the STEP physical file.

The STEP physical file is used by the second level process team. In this case, the Manufacturing team will use the two applications RAPIDS and MOSS, which are the constituent pieces of the MO prototype. Both RAPIDS and MOSS use the RAPIDS database. An interface, *STEP to RAPIDS,* facilitates translating the STEP physical file into a RAPIDS database. This is done by the C++ class library created by *express2c++* compiler.

The proposed MO system will use the STEP data directly. The purpose of providing the interfaces is to demonstrate the use of STEP physical files for information exchange. It is not a requirement for the top level team to use RAPIDS. The only requirement is that the top level team and the lower level teams are capable of creating and using the STEP physical file.

The Manufacturing Team passes back a consolidated position to the top level. To aid in the generation of a consolidated position, conflict resolution and design merging must be supported. This is done using the STEP Toolkit from STEP Tools Inc. The *diff* tool reads two versions of a design and creates a delta file. The *difference report generator* reads the difference file and the original design, and presents each STEP entity and its attributes with the original values and its change state clearly marked with an asterisks.

Once the conflicts of the Manufacturing team members have been resolved, design versions are merged using the STEP Tools *sed* tool. The *sed* tool read the delta file created by the *diff* tool and updates the original design version. This updated version of the design will be transferred back to the top-level product team as the Manufacturing Team's consolidated position.

The Requirements Manager is utilized by both the top level team and the manufacturing team in order to analyze the product design data against the specified manufacturability/producibility guidelines (refer to Section 5). At this time the RM is a stand

alone piece to the MO prototype, but future plans include directly tying the RM into the STEP data.

3.2 Results and Discussion

To illustrate how the prototype MO system can be utilized, we will step through an example of a PWB design originating from the top-level product team and being optimized for manufacturing by the Manufacturing Team using MOSS. The Project Coordination Board and the Requirements Manager, as they apply to the MO prototype environment, are detailed in Sections 4 and 5 of this report.

Shown in Figure 3-3 is a screen shot of the RAPIDS application with a PWB design loaded. The board profile, layer stackup, and initial component placement have been specified.

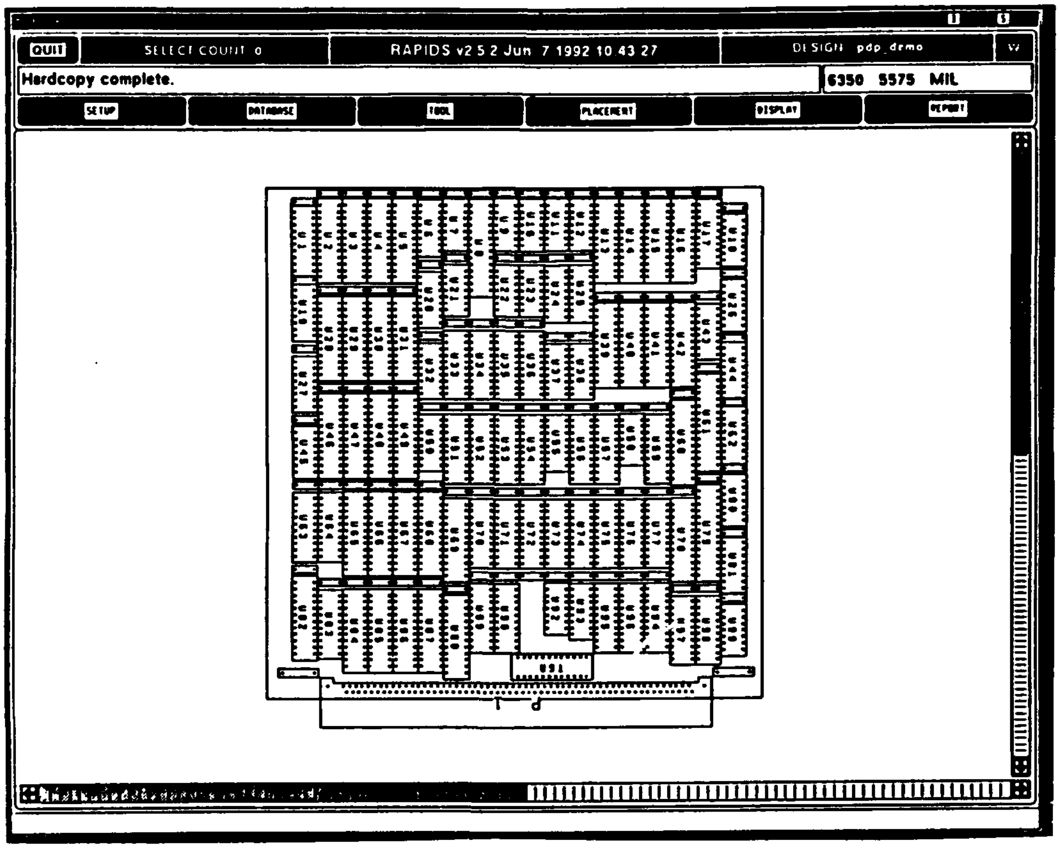

Figure 3-3. PWB Design Displayed in RAPIDS

The design is translated to STEP and is transitioned to the lower-level process teams for their input. For this enterprise the responsibilities of the Manufacturing Team are shared

between two process engineers. One will concentrate on optimizing the design for bare board fabrication, and the other will concentrate on optimizing for assembly. Since the tools that these engineers use run off of the RAPIDS database, the design will be translated to the RAPIDS database via the *step_to_rapids* interface.

The process engineer responsible for assembly performs an automatic insertion analysis in RAPIDS. Figure 3-4 illustrates the RAPIDS display showing one device highlighted with a thick border to indicate that the device must be manually inserted.

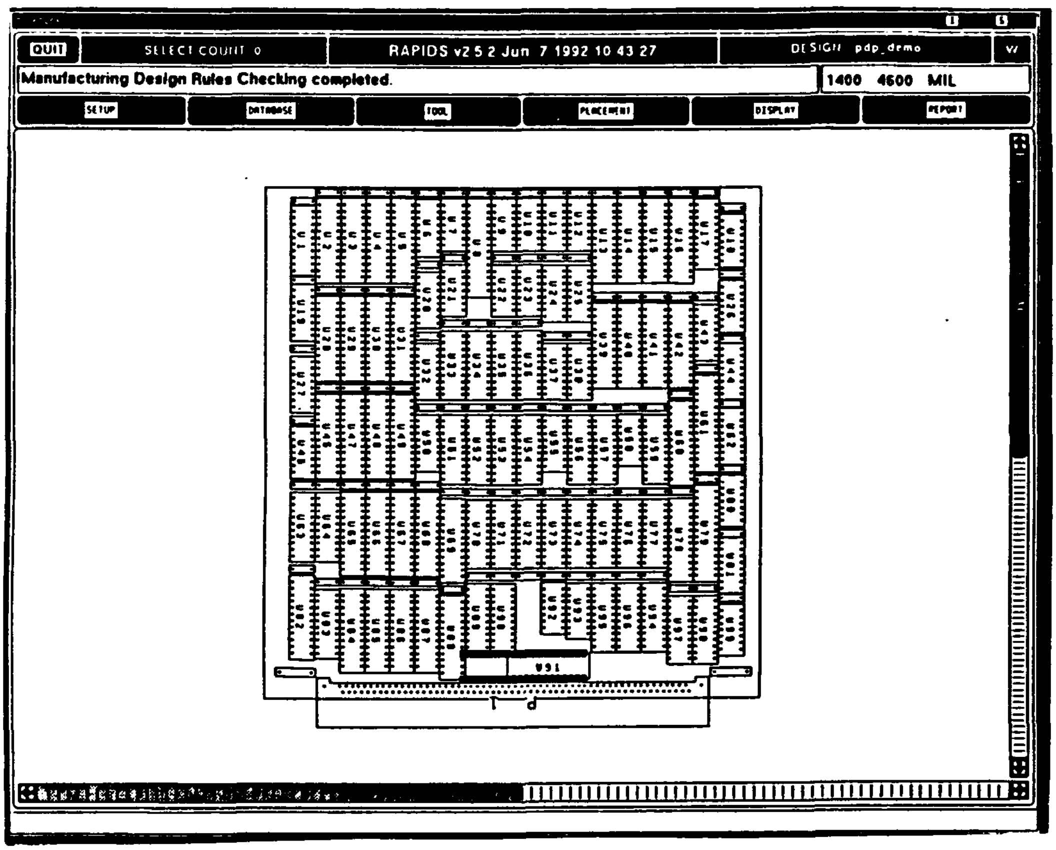

Figure 3-4. Device Requiring Manual Insertion

The problem occurred because devices U89 and U90 were obstructing the jaws of the insertion toolhead. By adjusting U91's location and rotation, the engineer can eliminate this problem and so he does so. The resulting design placement is shown in Figure 3-5.

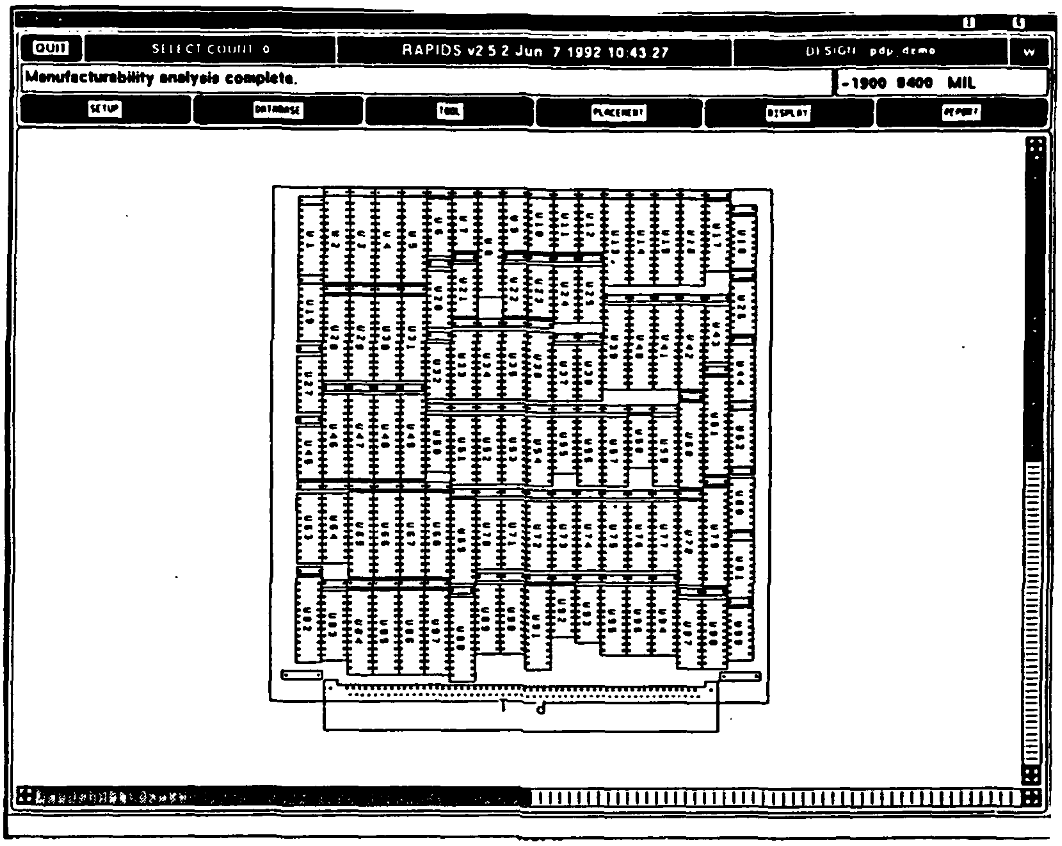

Figure 3-5. Insertion Problem Fixed

While the assembly process engineer is performing his automatic insertion analysis, the fabrication process engineer performs fabrication cost and yield. The tool he will use for this is MOSS. The engineer runs two analyses on the fabrication of the board. The first run has an end item conductor line width set to 10 mils, which represents the original value in the design, and in the second run has a value set to 5 mils. Figure 3-6 shows the MOSS comparison window with the results of each analysis run shown. As seen in the figure the fabrication yield and cost of the board is improved using a line width of 5 mils. The fabrication engineer changes the line width to 5 mils based on the analysis.

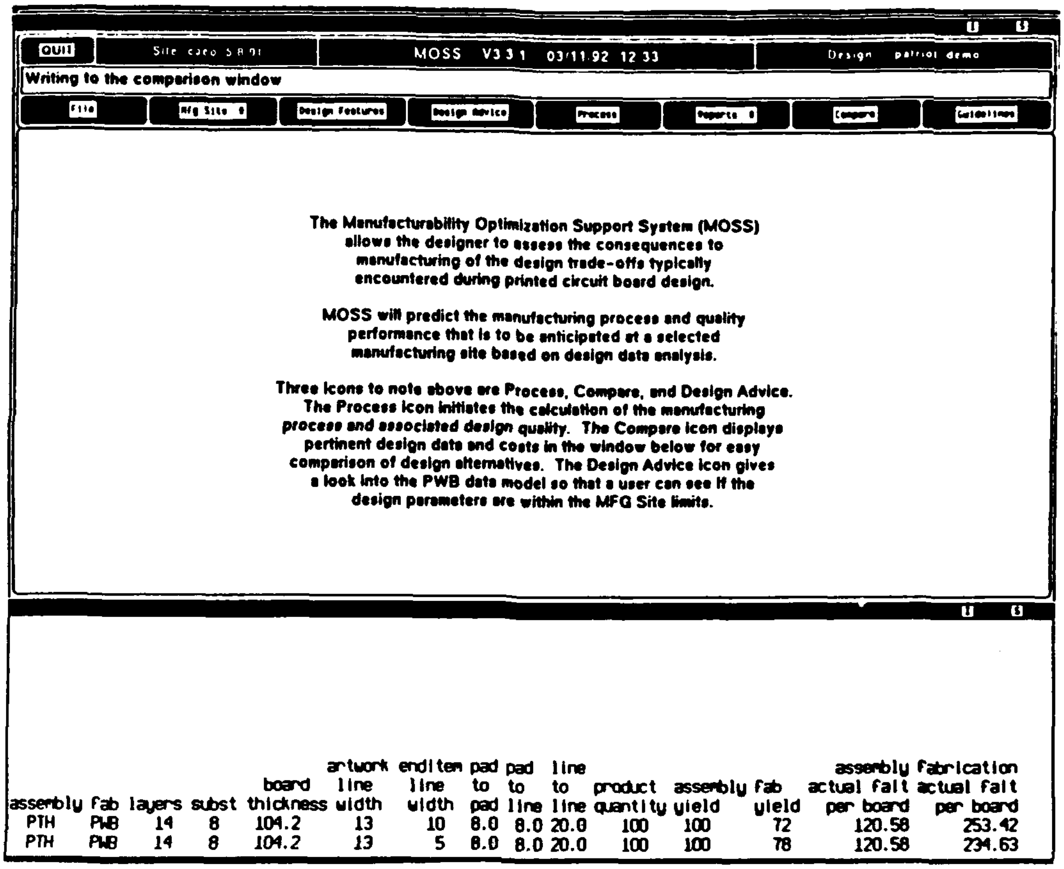

Figure 3-6. MOSS Comparison Window

Both manufacturing process engineers are now finished with their changes and must now pass their consolidated position back up to the top-level product team. Each of the engineers translates their RAPIDS designs back to STEP for conflict detection and transitioning back to the product team. Using the STEP *diff* tool creates a delta file, and runs the delta file through the *report generator*. A sample report showing the placement changes (changes are in bold) which the assembly engineer performed are provided below in Figure 3-7.

[xref_rec # 99024]	[xref_rec # 99024]	
symbolic: U91	symbolic: U91	
old_symbolic: U91	old_symbolic: U91	
model: ckr22.1	model: ckr22.1	
location: [point_rec # 99025]	location: [point_rec # 99025]	
x: 310000	x: 310000	
y: 150000	y: 150000	
mirror: 0	mirror: 0	
rotation: 270	**rotation: 0**	*
symbolic_flag: 0	symbolic_flag: 0	
external: 0	external: 0	

usa_device:
physical: M39014/22-1046
raytheon:
design_rules:
layer: TOP
via_flag: 0
location_set: E
auto_insert:
swap_inhibit: 0
fix: 0
device_bias: 0
thermal_bias: 0
coupling: <NULL>
decoupling: 0
space_rule: <NULL>
overlap: <NULL>
heat_sink:
power_dissip: 0
load_data: <NULL>
comment:
attribute: <NULL>

[xref_rec # 99024]
symbolic: C91
old_symbolic: C91
model: ckr22.1
location: [point_rec # 99025]
x: 310000
y: 150000
mirror: 0
rotation: 0
symbolic_flag: 0
external: 0
usa_device:
physical: M39014/22-1046
raytheon:
design_rules:
layer: TOP
via_flag: 0
location_set: E
auto_insert:
swap_inhibit: 0
fix: 0
device_bias: 0
thermal_bias: 0
coupling: <NULL>
decoupling: 0
space_rule: <NULL>
overlap: <NULL>
heat_sink:
power_dissip: 0
load_data: <NULL>

usa_device:
physical: M39014/22-1046
raytheon:
design_rules:
layer: TOP
via_flag: 0
location_set: E
auto_insert:
swap_inhibit: 0
fix: 0
device_bias: 0
thermal_bias: 0
coupling: <NULL>
decoupling: 0
space_rule: <NULL>
overlap: <NULL>
heat_sink:
power_dissip: 0
load_data: <NULL>
comment:
attribute: <NULL>

[xref_rec # 99024]
symbolic: C91
old_symbolic: C91
model: ckr22.1
location: [point_rec # 99025]
x: 305000 *
y: 125000 *
mirror: 0
rotation: 0
symbolic_flag: 0
external: 0
usa_device:
physical: M39014/22-1046
raytheon:
design_rules:
layer: TOP
via_flag: 0
location_set: E
auto_insert:
swap_inhibit: 0
fix: 0
device_bias: 0
thermal_bias: 0
coupling: <NULL>
decoupling: 0
space_rule: <NULL>
overlap: <NULL>
heat_sink:
power_dissip: 0
load_data: <NULL>

comment:	comment:
attribute: <NULL>	attribute: <NULL>

Figure 3-7. Difference Report Showing Placement Change

The changes do not conflict since one manufacturing engineer changed the placement of a component and the other changed the conductor line width that the board will be fabricated with. Since no conficts will arise from their changes, the designs can be merged together without any problems. The STEP *sed* tool is used to first merge the assembly engineers version with the Manufacturing Team's original version and then the fabrication process engineer's version is also merged into the original. Had there been any conflicts between two or more members of the process team, they would have to be resolved prior to merging the designs. The Manufacturing team now transitions their consolidated position in the form of a STEP physical file back up to the top-level product team.

At anytime during this design/analyze cycle, the process engineers can access the RM to verify that the requirements of the design are being met. Also, the project team manager/leader can utilize the PCB to see which tasks have been completed or are still being worked.

The MO prototype described above was developed to demonstrate the integration of established concurrent design and analysis systems in support of the two tiered team approach to the product/process development activities for PWB designs which requires design conflict detection and design change merging. In addition to those development efforts, we have been designing an approach for modeling manufacturing processes. The process modeling technique proposed for the MO system is being designed as an AND/OR dependency graph made up of selectable manufacturing processes, which can be either a process, operation, or step object. Each object in the model can be connected to object or objects at a higher and/or lower level in the graph. The difference between the object types is in the level of processing or planning detail defined (i.e. process decisions, operation planning, and/or detail operation planning). Two EXPRESS schemas, which represent the initial efforts on the MO process modeling design, can be found in Appendix II. The first schema (Appendix II – 10.1) is the MO process model. The second schema (Appendix II – 10.2) represents the valid grammar format for the MO process rules required for selecting the processes which are required to manufacture the part. An EXPRESS-G model of the MO process model schema is shown in Figure 3-8.

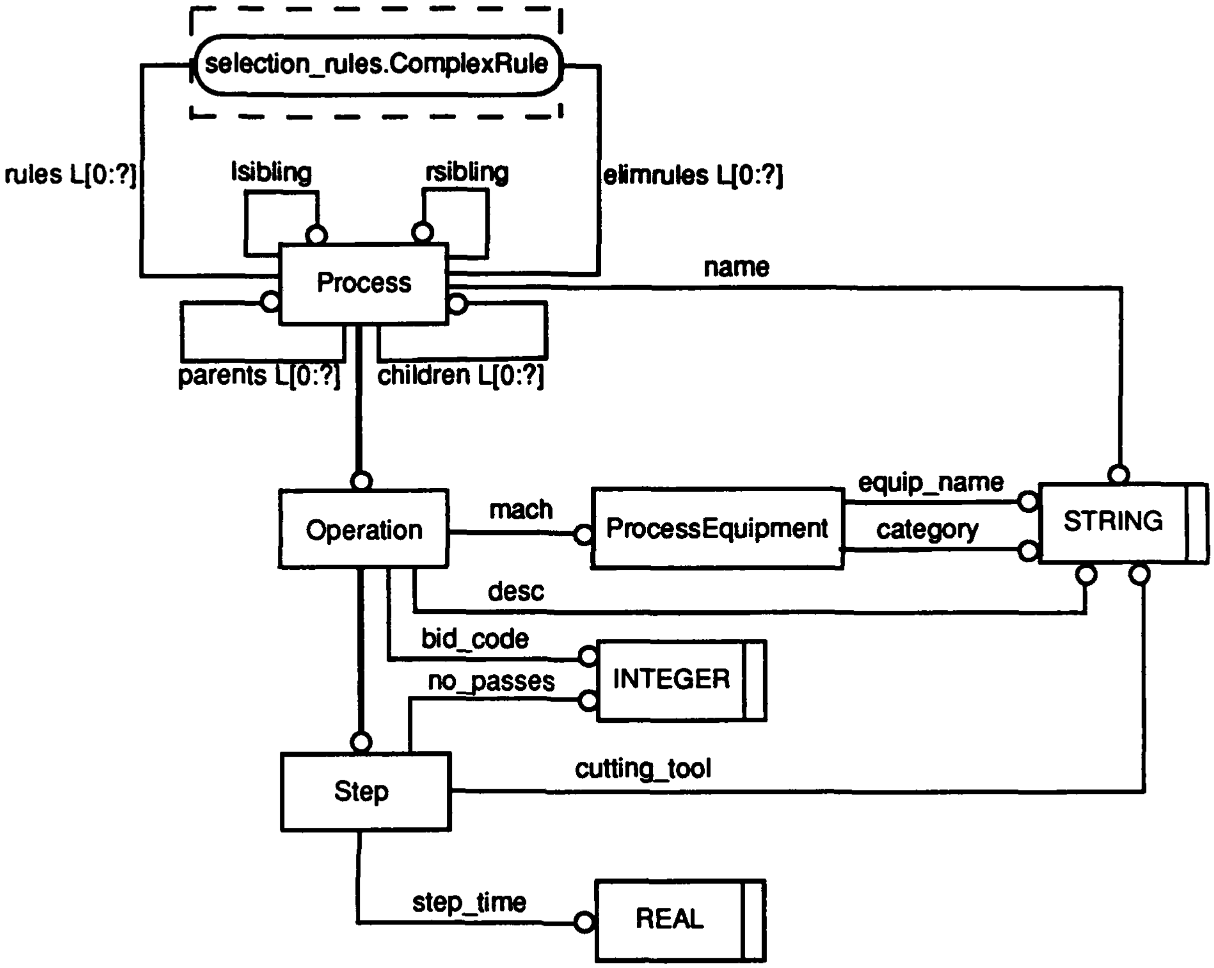

Figure 3-8. EXPRESS-G Representation of the MO Process Model Schema

4. PCB/CM - Project Coordination Board/ Communications Manager

4.1 Methods, Assumptions, and Procedures

4.1.1 Overview

The Project Coordination Board (PCB) is a system being developed to provide support for the coordination of the product development activities in a cooperative environment. The PCB provides common visibility and change notification through the common workspace, planning and scheduling of activities through the task structure, monitoring progress of product development through the product structure (i.e. constraints), and computer support for team structure through messages. The PCB is composed of two modules: the *cw* (Common Workspace) module and the *da* (Design Agent) module. The *cw* module provides the functionality for project coordination, and the *da* module is an interface provided to product developers so they can interact with the *cw* module. This tool is currently integrated with the Communications Manager.

The Communications Manager (CM) is a collection of modules that facilitates distributed computing in a heterogeneous network. It promotes the notion of a virtual network of resources which the project team members can exploit without any prior knowledge of the underlying physical network. The CM would be useful for those that would like to build transparent tools, virtual project networks, have access to remote tools, perform network tasks, perform message passing, and/or perform inter-process file transfers. The Communication and Directory Services provided in the CM module are required to utilize the PCB.

4.1.2 Proposed Use In MO

MO introduces the concept of a two tiered virtual tiger team. The two tiered approach consists of a cross functional product team linked to teams within each of the functions, in this case a manufacturing process team. To implement this approach there must be communication among the members of each team, and between the product and process team. We propose to use the PCB/CM to support the following capabilities which are required to support this communication:

- Product - to - Process Team Communication
  - Notification of design task completed or other pertinent status information.
  - Notification and issuance of database available for analysis.
  - Notification of alternative designs or trade-off decisions under consideration.

- Process - to - Product Team Communication
  - Notification and issuance of analysis results.
  - Notification and issuance of modified database with recommended changes.
  - Notification of changes to the process, guidelines, cost or yield models.

4.1.3 Evaluation Plan

Raytheon developed a task structure which modeled a sample printed wiring board (PWB) design cycle to demonstrate the applicability of the PCB to adequately support the product-to-process team communications. We modeled the design cycle using the task structure file format provided so that the Project Lead could initialize the task structure in one step. We then simulated the product-to-process team communication by stepping through the entire task structure. Detailed below are the actual steps that we performed to evaluated the PCB.

Step 1: Installed a copy of the Communications Manager (CM), and the Project Coordination Board (PCB).

Step 2: Started the Communication and Directory Services provided in the CM module (i.e. rdbinit)

Step 3: Ran the *cw* (Common Workspace) module to create a common workspace for the tutorial lesson.

Step 4: Performed the PCB tutorial lesson provided in the Training Script Manuals (References 5 and 6) to become familiar with PCB functionality and features.

Step 5: Ran the *cw* (Common Workspace) module again to create a common workspace for the PCB evaluation test case.

Step 6: Modeled a sample PWB design cycle using the task structure file format (refer to Appendix I).

Step 7: Initialized the task structure with the PWB design cycle model.

Step 8: Simulated the product-to-process team communications by stepping through the entire task structure.

4.2 Results and Discussion

We installed the PCB and CM Modules that we received from CERC, and performed the tutorial lesson provided in the PCB Training Script Manuals (References 5 and 6). This provided us with an understanding of the PCB functionality. After stepping through the training script, we developed a sample task structure to test the applicability of the PCB to adequately model the product-to-process communication procedures (refer to Appendix I). This task structure modeled a sample PWB design cycle. Included are major design steps, such as concept development, design capture, design verification, component placement, routing, transition to production, and several design reviews. The design reviews included representatives from design, test, reliability, manufacturing, and thermal. Following is a high level view which represents the design cycle steps which we modeled.

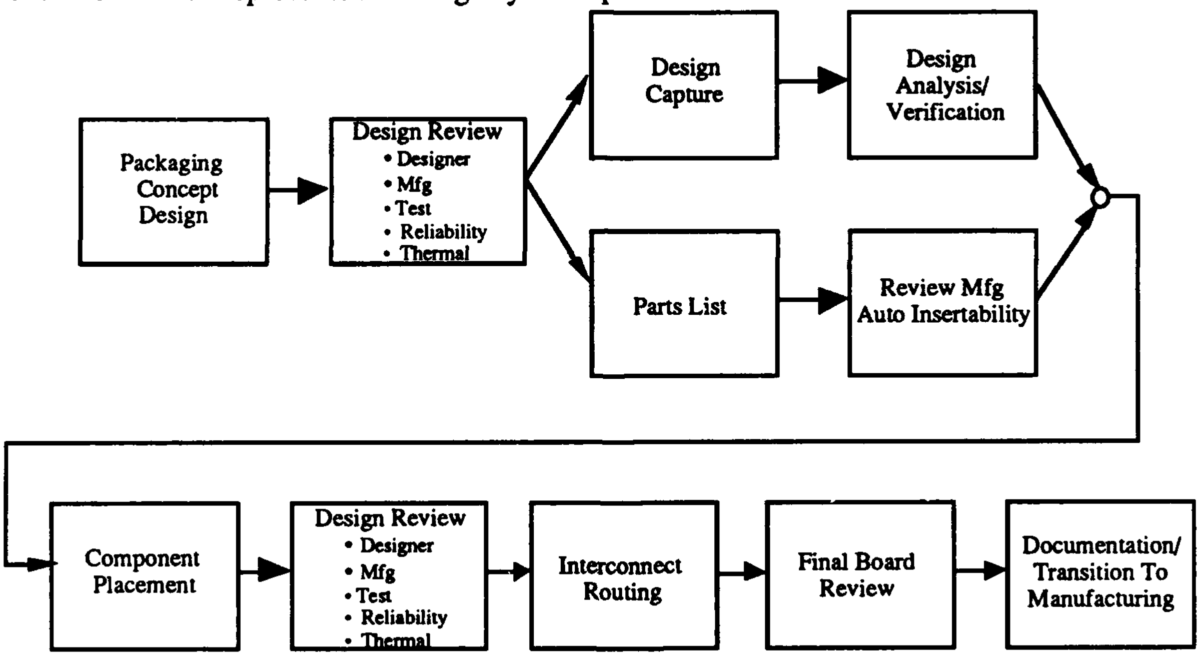

Figure 4-1. Sample PWB Design Cycle Flow

The Project Lead (user with special privileges) initialized the product task structure from a file (Appendix I). The Project Lead could then view any task or work order that appeared in the network, add a task to the existing network, acknowledge receiving a task, and indicate completion of a task. There was no way of attaching another file (i.e. an enclosure as in EMail) to a message or work order. The other users could acknowledge receiving a task and indicate

completion of that task. The PCB automatically dispatched tasks as previous tasks were completed, as well as, the Project Lead could dispatch a task.

The product data structure could be initialized through a file by the Project Lead. Once the product data structure was loaded, it was displayed using a hierarchical browser. The PCB allowed the user to view (objects, constraints, and assertions), edit (object, attribute, and value), assert (values, constraints, and profiles), and save, load, or clear the knowledge base (*kb*). When the users were logged out of the PCB, messages concerning the release of new product models, constraint violations on attributes which users have set a perspective, and any tasks that had been assigned were maintained for them and could be viewed at their convenience. Since the product data structure in the PCB knowledge base (*kb*) provided only minimal min/max constraint value checking, we found no need for the product data structure at this time. It is also extremely cumbersome to enter your product data into the PCB knowledge base (*kb*) and impossible to get your modified data out to restore in our own data repository (i.e. ROSE).

Learning the basics of the PCB with the aid of the user and training script manuals provided with the software (References 5, 6, and 7) was straightforward. Based on the results of our sample test case, we believe that the PCB is able to handle only simple task structures, since the only reliable and complete way to get a task structure into the system is through the use of the flat file format which is cumbersome. The user interface data entry method stores the data directly in the *kb*, and there is no way to restore the task structure to its initial state once the process has started. The *cw* process is suppose to remain up and running when the Project Lead logs off the computer system so that the other users could have access to the common workspace, but during our evaluation it kept exiting when the Project Lead logged off. This problem is suppose to be fixed under X11R5. We were and are still running X11R4, and the only workable solution that was offered was to upgrade to X11R5. At this time, we still feel it is not mature enough and lacks the functionality required for real-world environments. Refer to Table 4-1 for a summary of the evaluation results of the system.

Table 4-1. Summary of the PCB/CM Evaluation Results

System Under Evaluation	MO Applicability	Maturity/Availability	Conclusion/ Recommendation
PCB	Support communication of product/process development activities.	Unstable due to 'cw' problems. Not sophisticated enough to model complex task structures.	Lacks the functionality required for real-world environment. Found no current use for the product data structure portion. Recommend using only for a simple two tiered approach to the product/process development activities.
CM	Support communication among team members. Support distributed computing & database access in a network.	Stable. No difficulty with the installation procedures.	Recommend using the Communication and Directory Services portion of the CM module since it is required for the PCB.

5. RM - Requirements Manager

5.1 Methods, Assumptions, and Procedures

5.1.1 Overview

The Requirements Manager (RM) is a software tool designed to manage product requirements and evaluate the compliance of product design data with requirements. The tool allows the user to 1) model requirements or guidelines, 2) model the product design data structure, 3) populate the product design data structure with product data and 4) evaluate to what extent the product design data meets the specified requirements. As a result of the evaluation process, the tool will provide the user with a status (Pass, Fail) of the compliance of the product data with the requirements.

5.1.2 Proposed Use In MO

The purpose of integrating the MO manufacturing guideline functionality into the RM is to provide the "top level" product development team insight into manufacturing requirements apart from the MO analyses.

It is common practice for a manufacturer to document manufacturability, or producibility guidelines that delineate standard manufacturing practices and acceptable design parameters. The purpose of these guidelines is to communicate the capabilities of the manufacturing process to the product design community to ensure that new product designs are specified within manufacturing capabilities. The guidelines delineate quantitative and qualitative producibility issues.

One of the functions of MO is to provide evaluation of manufacturing guidelines. For each guideline entry there is a related recommendation. The guidelines can be evaluated separately, or triggered based on the process analysis module within the MO system. Unlike the process selection constraints, manufacturability guideline violations may not cause alternative selection. The result could be an operation cost increase, for instance, the need for non-standard tooling, a yield loss, or a less tangible impact. These guidelines will also be entered into the Requirements Manager so that they are available to the product design team along with the other requirements placed on the design.

5.1.3 Evaluation Plan

In order to evaluate the applicability of the RM tool to adequately model manufacturing guidelines, a test case was developed. The test case consisted of a set of selected manufacturing guidelines along with the appropriate product data structure. Once the guidelines and product data structure were modeled, the product data structure was populated and then evaluated against the guidelines. This was repeated several times with different sets of data. Detailed below are the actual steps that we performed to evaluate the RM.

Step 1: Installed a copy of RM release 3.0 on a 486 PC at Raytheon. This step was performed by Cimflex personnel.

Step 2: Received training and a demonstration from Cimflex Teknowledge.

Step 3: Modeled a select set of Raytheon manufacturing guidelines.

Step 4: Modeled the product data structure.

Step 5: Populated the product data structure with design data.

Step 6: Evaluated the compliance of the product data with the guidelines.

Step 7: Reviewed the status of each requirement.

Step 8: Repeated steps 5 through 7 with different data sets.

5.2 Results and Discussion

The Requirements Manager (RM) is a system designed to manage product requirements, specifications and corporate policies to support concurrent engineering. Within the MO program, Raytheon plans to use the RM to manage manufacturability and producibility guidelines and evaluate product design data for compliance with those guidelines.

After Cimflex installed the software and trained the evaluators, the evaluation of the product took approximately two weeks. A sample set of manufacturing guidelines to be modeled in the RM were chosen. Some examples of these guidelines are: "The maximum board dimension must be less than 14 inches", "Switches must be hermetically sealed", or "If the number of leads is less than or equal to 24 the span should be 0.3 inches". Each of these guidelines were entered into the "Requirement Flowdown Window". Details describing the requirement (guideline) were entered into the "Requirement Detail Window". A valuable feature of the RM tool is the ability to reconfigure the "Requirement Flowdown Window". This allows the

engineer to enter, edit and view only data that is pertinent at that given time. Another good feature of the "Requirement Flowdown Window" is the ability to arrange, edit and view the requirements in a hierarchical format. This format can be collapsed and expanded depending on how much detail the engineer needs to see.

The next step was to model the PWB product structure tree. The product structure tree is similar to a generation breakdown in which the hierarchy of the product is successively defined. The product structure tree is entered into the "Product Structure Window". Details describing each product element are entered into the "Product Element Detail Window". Examples of product elements that were entered are: "Printed Wiring Board" and "Component". Similar to the "Requirement Flowdown Window", the "Product Structure Window" has a hierarchical format and is reconfigurable — both strong features of the tool. For each element in the product structure tree, attributes can be defined. These attributes are entered into the "Attribute Detail Window". Examples of attributes which were entered are: "PWB length", "lead length", and "axial body diameter".

Describing the product elements and their associated attributes was awkward. This was due to a PWB consists of many Components *all with similar attributes*. In order to model the components as the tool exists, a separate product element must be entered for each component on the PWB. An improvement to the tool could be to create a product element library complete with attributes capable of being parameterized. With this feature the engineer could instantiate library elements and input those directly into the product element structure. In order to work around this deficiency of the tool, one "component" element was created. The component element was then populated once for each component on the PWB.

For each requirement entered into the requirement hierarchy structure, the user can associate an executable condition. The executable condition is an English-like expression which tests product data for compliance with the requirement (guideline in this case). An example of an executable condition is : "(("Baseplate thickness" of "PWB") lt 0.25)". This executable conditions says that the "Baseplate thickness" attribute of the "PWB" product element should be less than 0.25. Logical expressions, quantitative expressions, and qualitative expressions were entered and tested as part of the evaluation. This was found to be a powerful feature of the tool.

The next step was to populate the product element structure with data. The tool provides two documented methods (refer to Reference 8) to populate the data. The first method is to enter the data into the "Attribute Detail Window". The second method is to proceed and evaluate the requirement and let the tool automatically prompt you for the missing data. As part of the evaluation, both methods were tested. As an alternative, it would be more useful to populate the product element structure automatically from existing design database through direct access to the RM database. The RM User Manual (reference 8) stated that we could write an interface that would be able to access the RM's database directly. This methodology was not documented in the User Manual, it only had a note to call Cimflex Teknowledge about the precise approach. Preferably, the precise approach should be described in the User Manual.

The next step was to evaluate the design data for compliance with the guidelines and review the compliance status. This step involves selecting the "Evaluate" button on the "Requirement Flowdown Window". If the evaluation expression needs data which has not yet been entered, the tool will prompt the user to enter the data. (The user can also, for each attribute, tell the tool to prompt the user for the data regardless of whether or not the data was entered.) When the evaluation is complete, the tool will provide the user with the compliance status — Pass, Fail, Uncertain, Irrelevant, and Untested. In the case of a failed requirement, the user can view a description of why the requirement failed by calling up the "Status Explanation" field from the "Requirement Detail Window". In a similar fashion, rationale for a guideline can be viewed by calling up the "Rationale" field from the "Requirement Detail Window". (The rationale is not automatically generated , it must be input).

Another good feature of the tool is the ability to relate requirements to product elements and product attributes. This feature can be further enhanced by automatically creating the relationship when entering product elements and attributes into the "Executable Conditions" field of the "Requirement Detail Window".

Based on the results of our evaluation, we believe that the RM is able to provide the capabilities needed by MO for testing product design data for compliance to manufacturing guidelines. Refer to Table 5-1 for a summary of the evaluation results of the system.

Table 5-1. Summary of the RM Evaluation Results

System Under Evaluation	MO Applicability	Maturity/Availability	Conclusion/ Recommendation
RM	Manage manufacturability/ producibility guidelines.	Stable. No difficulty with installation or evaluation procedures.	Plan on tying the RM product data directly into the PWB STEP data.

6. Conclusions

During this reporting period, an initial MO prototype was developed which demonstrated a first pass at MO functionality, and evaluations of the Project Coordination Board and the Requirements Manager were performed. Table 6-1 contains the summary of the PCB/CM and RM evaluation results.

The prototype was built by utilizing two existing Raytheon developed systems, RAPIDS–Raytheon Automated Placement and Interconnect System and MOSS–Manufacturing Optimization Support System, the STEP Toolkit (including the ROSE DB system) from STEP Tools Inc., the Requirements Manager from Cimflex Teknowledge, and the Project Coordination Board from CERC. The prototype demonstrates concurrent design, concurrent analysis, design conflict detection, and design change merging of PWB designs. The functionality in the prototype include the RAPIDS to ROSE translator, the delta file and design merging capabilities of ROSE, a generic difference report generator, and a printed circuit board design flow (task structure) modeled in the Project Coordination Board (PCB). A select set of manufacturing guidelines were modeled in the Requirements Manager (RM) as a standalone application.

Based on the results of our sample test case, we believe that the PCB is able to handle only simple task structures, since the only reliable and complete way to get a task structure into the system is through the use of the flat file format which is cumbersome. The user interface data entry method stored the data directly in the *kb*, and there was no way to restore the task structure to its initial state once the process had started. The *cw* process is suppose to remain up and running when the Project Lead logs off the computer system so that the other users could have access to the common workspace, but during our evaluation it kept exiting when the Project Lead logged off. This problem is suppose to be fixed under X11R5. We were and are still running X11R4, and the only workable solution that was offered was to upgrade to X11R5. At this time, we still feel it is not mature enough and lacks the functionality required for real-world environments.

Based on the results of our evaluation of the RM, we believe that it is able to provide the capabilities required by MO for testing product design data for compliance to manufacturing

guidelines. We plan on continuing the integration efforts of the RM System into the MO environment.

Raytheon will continue development of MO during the next quarter based on the initial prototype efforts for the Manufacturing Optimization (MO) System developed during the reporting period. Raytheon is also in the process of developing the Design Specification which will be delivered during the next quarter.

Table 6-1. Summary of the DICE Tool Evaluations

System Under Evaluation	MO Applicability	Maturity/Availability	Conclusion/ Recommendation
PCB	Support communication of product/process development activities.	Unstable due to 'cw' problems. Not sophisticated enough to model complex task structures.	Lacks the functionality required for real-world environment. Found no current use for the product data structure portion. Recommend using only for a simple two tiered approach to the product/process development activities.
CM	Support communication among team members. Support distributed computing & database access in a network.	Stable. No difficulty with the installation procedures.	Recommend using the Communication and Directory Services portion of the CM module since it is required for the PCB.
RM	Manage manufacturability/ producibility guidelines.	Stable. No difficulty with installation or evaluation procedures.	Plan on tying the RM product data directly into the PWB STEP data.

7. References

1. BR-20558-1, 14 June 1991, DARPA Initiative In Concurrent Engineering (DICE) Manufacturing Optimization - Volume I - Technical.

2. CDRL No. 0002-AC-1, 19 March 1992, Operational Concept Document For The Manufacturing Optimization (MO) System. Contract No. MDA972-92-C-0020.

3. CDRL No. 0002-AC-2, 19 March 1992, Description of Concurrent Engineering Technology For The Manufacturing Optimization (MO) System. Contract No. MDA972-92-C-0020.

4. CDRL No. 0002-AC-3, 31 May 1992, Functional Requirements and Measure of Performance For The Manufacturing Optimization (MO) System. Contract No. MDA972-92-C-0020.

5. Project Coordination Board (PCB) Training Script for The Project Lead Role.

6. Project Coordination Board (PCB) Training Script for The User.

7. User Manual for the Project Coordination Board (PCB) of DICE (DARPA Initiative in Concurrent Engineering), July 10, 1992.

8. ProductTrack Requirements Manager User Guide and Reference, Release 1.01 for Microsoft Windows and Gupta SQLBase, CIMFLEX Teknowledge Corporation, July, 1992.

8. Notes

8.1 Acronyms

CAEO	Computer Aided Engineering Operations
CDRL	Contract Data Requirements List
CM	Communications Manager
DARPA	Defense Advanced Research Projects Agency
DFMA	Design for Manufacturing and Assembly
DICE	DARPA Initiative In Concurrent Engineering
MO	Manufacturing Optimization
PCB	Project Coordination Board
PWB	Printed Wiring Board
RAPIDS	Raytheon Automated Placement and Interconnect Design System
ROSE	Rensselaer Object System For Engineering
RM	Requirements Manager

Appendix I - PCB Task Structure File

{pdpdemo
sender: "Project_Lead"
due_date: "01 10 92"
earliest_start: "92 92 91"
destination: "Project_Lead"
description: "Start PDP PWB Task Structure"
focus: "pdp"
previous_tasks * (next_tasks) :
next_tasks * (previous_tasks):task1_Design1_define_concept
}
{task1_Design1_define_concept
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Design1"
description: "Define Packaging Concept"
focus: "pdp"
output: "Concept Package"
previous_tasks * (next_tasks) : pdpdemo
next_tasks * (previous_tasks):task2_Design_review task2_Mfg_review task2_Test_review task2_Reliab_review task2_Thermal_review
}
{task2_Design_review
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Design1"
description: "Review Concept Functionality"
focus: "pdp"
previous_tasks * (next_tasks) : task1_Design1_define_concept
next_tasks * (previous_tasks): PLead_final_concept_review
}
{task2_Mfg_review
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Mfg"
description: "Review Concept Manufacturability"
focus: "pdp"
previous_tasks * (next_tasks) : task1_Design1_define_concept
next_tasks * (previous_tasks): PLead_final_concept_review
}
{task2_Test_review
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Test"
description: "Review Concept Testability"
focus: "pdp"
previous_tasks * (next_tasks) : task1_Design1_define_concept
next_tasks * (previous_tasks): PLead_final_concept_review
}
{task2_Reliab_review

sender: "Project_Lead"
due_date: "01 10 92"
destination: "Reliability"
description: "Review Concept Reliability"
focus: "pdp"
previous_tasks * (next_tasks) : task1_Design1_define_concept
next_tasks * (previous_tasks): PLead_final_concept_review
}
{task2_Thermal_review
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Thermal"
description: "Review Concept Fit, Power, and Thermal"
focus: "pdp"
previous_tasks * (next_tasks) : task1_Design1_define_concept
next_tasks * (previous_tasks): PLead_final_concept_review
}
{PLead_final_concept_review
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Project_Lead"
description: "Check Concept Review"
focus: "pdp"
previous_tasks * (next_tasks):task2_Design_review task2_Mfg_review task2_Test_review task2_Reliab_review task2_Thermal_review
next_tasks * (previous_tasks): task3_Design1_schematic_capture task3_Design2_parts_list
}
{task3_Design1_schematic_capture
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Design1"
description: "Input Engineering Schematic"
focus: "pdp"
previous_tasks * (next_tasks) : PLead_final_concept_review
next_tasks * (previous_tasks): task4_Design2_verification
}
{task3_Design2_parts_list
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Design2"
description: "Define Parts List"
focus: "pdp"
previous_tasks * (next_tasks) : PLead_final_concept_review
next_tasks * (previous_tasks):task4_Mfg_review_insertability
}
{task4_Design2_verification
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Design2"
description: "Perform Simulation and Load Analysis"
focus: "pdp"
previous_tasks * (next_tasks) : task3_Design1_schematic_capture
next_tasks * (previous_tasks):task5_Design1_comp_placement

}
{task5_Design1_comp_placement
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Project_Lead"
description: "Perform Component Placement"
focus: "pdp"
previous_tasks * (next_tasks) : task4_Design2_verification task4_Mfg_review_insertability
next_tasks * (previous_tasks): task6_Thermal_comp_placement_review
task6_Mfg_comp_placement_review
}
{task4_Mfg_review_insertability
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Mfg"
description: "Review parts list for auto-insertion considerations."
focus: "pdp"
previous_tasks * (next_tasks) : task3_Design2_parts_list
next_tasks * (previous_tasks): task5_Design1_comp_placement
}
{task6_Thermal_comp_placement_review
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Thermal"
description: "Perform Thermal Analysis on board placement."
focus: "pdp"
previous_tasks * (next_tasks) : task5_Design1_comp_placement
next_tasks * (previous_tasks): task7_Design2_rout
}
{task6_Mfg_comp_placement_review
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Mfg"
description: "Perform Producibility Analysis (auto-insertability considerations) on board placement."
focus: "pdp"
previous_tasks * (next_tasks) : task5_Design1_comp_placement
next_tasks * (previous_tasks): task7_Design2_rout
}
{task7_Design2_rout
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Design2"
description: "Perform Routing on Board."
focus: "pdp"
previous_tasks * (next_tasks) : task6_Mfg_comp_placement_review
task6_Thermal_comp_placement_review
next_tasks * (previous_tasks): PLead_final_board_review
}
{PLead_final_board_review
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Project Leader"

description: "Perform Final Board Review."
focus: "pdp"
previous_tasks * (next_tasks) : task7_Design2_rout
next_tasks * (previous_tasks): PLead_Transition_to_Production
}
{PLead_Transition_to_Production
sender: "Project_Lead"
due_date: "01 10 92"
destination: "Project Leader"
description: "Transition PDP Board to Manufacturing."
focus: "pdp"
previous_tasks * (next_tasks) : PLead_final_board_review
next_tasks * (previous_tasks):
}

Appendix II - Process Modeling Techniqu

10. MO Process Dependency Model

10.1 MO Process Modeling Express Schema

```
-------------------------------------------------------------------------------
-- MO Process Modeling Schema
-- Author:  L.J.Lapointe
-- Description: This schema is model to support modeling of
                  manufacturing Processes using an AND/OR
                dependency graph made up of selectable
                  manufacturing processes, which can be either a
                  process, operation, or step object. Each object in
                  the model can be connected to object(s) at a higher
                  level and/or lower level in the graph. The difference
                  between the object types is in the level of processing
                  (planning) detail (i.e. process decisions, operation
                  planning, and/or detail operation planning).
-------------------------------------------------------------------------------

INCLUDE 'rules.exp';                         -- Selection/Elimination Rules Format

SCHEMA process_model;                        -- MO Process Dependency Model

REFERENCE FROM selection_rules;

  ENTITY ProcessEquipment;
    equip_name: STRING;                      -- Equipment Name
    category: STRING;                        -- Equipment Category
  END_ENTITY;

  ENTITY Process;
    name: STRING;                            -- Process Name
    rules: LIST [0:?] OF ComplexRule;        -- List of Process Selection Rules
    elimrules: LIST [0:?] OF ComplexRule;    -- List of Process Elimination Rules
    parents : LIST [0:?] OF Process;         -- List of Parents (Ancestors)
    children : LIST [0:?] OF Process;        -- List of Children (Descendents)
    lsibling : Process;                      -- Left Sibling
    rsibling : Process;                      -- Right Sibling
  END_ENTITY;

  ENTITY Operation
    SUBTYPE OF (Process);                    -- Operation Inherited from Process
      desc: STRING;                          -- Description
      mach: ProcessEquipment;                -- Process Equipment
      bid_code: INTEGER;                     -- Bid Code
  END_ENTITY;

  ENTITY Step
    SUBTYPE OF (Operation);                  -- Step Inherited from Operation
```

```
    cutting_tool: STRING;             -- Cutting Tool
    no_passes: INTEGER;               -- Number of Passes
    step_time: REAL;                  -- Step Time
  END_ENTITY;

END_SCHEMA;
```

10.2 MO Selection Rules Express Schema

```
------------------------------------------------------------------
-- Selection Rules Schema
-- Author: L.J.Lapointe
-- Description: This schema models the Grammar format for
                MO Process Selection and Elimination Rules.
------------------------------------------------------------------

SCHEMA selection_rules;

  CONSTANT
   Multiply : STRING := '*';
   Divide   : STRING := '/';
   Add      : STRING := '+';
   Subtract : STRING := '-';

   U_Op     : STRING := '!';

   Less         : STRING := '<';
   LessEqual    : STRING := '<=';
   Greater      : STRING := '>';
   GreaterEqual : STRING := '>=';
   Equal        : STRING := '=';
   NotEqual     : STRING := '!=';

   LP        : STRING := '(';
   RP        : STRING := ')';
   Comma     : STRING := ',';
   DQ        : STRING := '"';
  END_CONSTANT;

  TYPE DQuote = ENUMERATION OF
        (DQ);
  END_TYPE;

  TYPE AND_Op = ENUMERATION OF
        (Comma);
  END_TYPE;

  TYPE LParen = ENUMERATION OF
        (LP);
  END_TYPE;
```

```
TYPE RParen = ENUMERATION OF
        (RP);
END_TYPE;

TYPE Unary_Op = ENUMERATION OF
        (U_Op);
END_TYPE;

TYPE Real_numbers = REAL;
END_TYPE;

TYPE Integers = INTEGER;
END_TYPE;

TYPE
  Const = SELECT (Real_numbers, Integers);
END_TYPE;

TYPE Operator = ENUMERATION OF
    (Multiply, Divide, Add, Subtract);
END_TYPE;

TYPE Equiv_Op = ENUMERATION OF
   (Less, LessEqual, Greater, GreaterEqual, Equal, NotEqual);
END_TYPE;

TYPE
  Equation = SELECT (Term, ComplexEquation);
END_TYPE;

ENTITY DataDictStr;
  dstr : STRING;
END_ENTITY;

ENTITY ComplexEquation;
 Var1    : Term;
 Oper1   : Operator;
 Value   : Equation;
END_ENTITY;

ENTITY ParenEquation;
 Lparenthesis : LParen;
 Equ          : Equation;
 Rparenthesis : RParen;
END_ENTITY;

TYPE
 Term = SELECT (Const, DataDictStr, ParenEquation);
END_TYPE;

TYPE
 Expression = SELECT (Equation, ComplexExp, SimpleExp, StringValue);
END_TYPE;
```

```
  ENTITY StringValue;
   quote1 : DQuote;
   value1 : STRING;
   quote2 : DQuote;
  END_ENTITY;

  ENTITY ComplexExp;
   Equ1     : Equation;
   EquivOp1 : Equiv_Op;
   Exp1     : Expression;
  END_ENTITY;

  ENTITY SimpleExp;
   Not1 : Unary_Op;
   DataDictVar : DataDictStr;
  END_ENTITY;

  ENTITY Rules;
   exp1 : Expression;
   And1 : And_Op;
  END_ENTITY;

  ENTITY ComplexRule;
   lrule: LIST [0:?] OF Rules;
  END_ENTITY;

END_SCHEMA;
```

Distribution List

DPRO-Raytheon
C/O Raytheon Company
Spencer Lab., Wayside Ave.
(one copy of each report)

Defense Advanced Research Projects Agency
ATTN: Defense Sciences Office; Dr. H. Lee Buchanan
Virginia Square Plaza
3701 N. Fairfax Drive
Arlington, VA. 22203-1714
(one copy of each report)

Defense Advanced Research Projects Agency
ATTN: Electronic Systems Technology Office; Capt. Nicholas J. Naclerio, USAF
Virginia Square Plaza
3701 N. Fairfax Drive
Arlington, VA. 22203-1714
(one copy of each report)

Defense Advanced Research Projects Agency
ATTN: Contracts Management Office; Mr. Donald C. Sharkus
Virginia Square Plaza
3701 N. Fairfax Drive
Arlington, VA. 22203-1714
(one copy of each report)

Defense Technical Information Center
Building 5, Cameron Station
ATTN: Selections
Alexandria, VA 22304
(two copies of each report)